Steepness Is Not Selectivity: Why Post-hoc Unlearning Misses the Retraining Direction

Anonymous authors
Paper under double-blind review

Abstract

Machine unlearning is usually framed as a search for a better objective. That framing rests on a premise that is rarely tested: that the update retraining would make is one that a post-hoc procedure can find. We test it directly. Using as an oracle a model retrained without the forget set from the same base, we find the obstacle is not entanglement: the straight path to that oracle is almost perfectly selective — on our primary family and removal size forget loss rises 140× while retain loss rises by at most 1.19× and ends *below* where it started, and the picture survives when forget and retain concern the *same* entities — yet standard unlearning updates that reach a comparable forget level are orthogonal to it, at cos ≈ 0.009 in a space where two runs of the same method aimed at *disjoint* fact sets align at 0.035. We identify one mechanism: a separation between *steepness*, forget loss gained per unit of weight movement, and *selectivity*, forget loss gained per unit of retain damage: Euclidean steepest ascent on the forget loss maximizes the first, unlearning wants the second, and nothing ties them together. Plain SGD, where that argument applies literally, is steeper still and barely selective, so the optimizer people actually use is a partial remedy rather than a confound. Preconditioning ascent by a diagonal retain-Fisher proxy for that curvature instead, as the mechanism prescribes, cuts collateral damage by an order of magnitude and moves *further* from the oracle direction: the local optimum of selectivity is not the retraining direction. Two further results say the premise fails even if the search succeeded. Across the four methods tested, all near-orthogonal (cos ≤ 0.065), better alignment does not predict better selectivity — weight arithmetic is 7–8× closer to the oracle and no more selective. And the target does not transfer: measured from a different full-data solution the retraining displacement is nearly orthogonal (cos = 0.029) to the one measured here. The existence claim holds in three model families spanning 0.5–1.5B, and the geometry in the two where we ran the full comparison.

1 Introduction

Machine unlearning asks for a model that behaves as though some subset of its training data had never been seen. The definition is operational — retrain without that subset — and retraining is exactly what nobody can afford, so the field builds post-hoc procedures that edit a trained model instead (Eldan & Russinovich, 2023; Kurmanji et al., 2023). It is by now well documented that these procedures pay for forgetting with collateral damage: suppressing the forget set degrades retained knowledge, and the suppression is often shallow enough to be undone (Maini et al., 2024; Zhang et al., 2024). Almost all of the response has been to design a better objective.

One natural explanation for why that is hard is *entanglement*: the parameters that carry the forget set are the parameters that carry everything else, so no edit can separate them. It is the premise behind the projection-based remedies that have appeared recently (Shamsian et al., 2025; Wang et al., 2025; Zhou et al., 2025), which build an update orthogonal to the

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

retain gradients precisely because the two are assumed inseparable otherwise, and it is what the forget–retain trade-off curves in the benchmark literature (Maini et al., 2024; Shi et al., 2025) are usually read as showing. That explanation is testable, and it has a natural test object. Retraining without F is not just the definition of success — it is a specific point in weight space, and the vector from the trained model to it, $\Delta_{\text{oracle}} = \theta(M_{\setminus F}) - \theta(M_0)$, is a direction we can measure against. If entanglement were the obstacle, that direction would itself be non-selective: reaching the retrain endpoint would have to cost retained performance along the way.

It does not. The straight path from $\theta(M_0)$ to the retrain oracle raises forget-set negative log-likelihood by 140× while retain loss rises by at most 1.19× and ends *below* where it started, with no barrier in between (§3). A mirror control confirms the direction is specific to F, and a norm-matched seed-only difference of the same length moves forget loss by 14% rather than 140×, so neither the specificity nor the effect is an artifact of how far the model moved. The selective direction exists. Our splits are disjoint fictitious entities, so what this refutes is entanglement between *independent facts* — the setting in which retraining is the definition of success rather than a proxy; where forget and retain share features such a direction need not exist (§8).

What fails is the *search* for it. Three standard unlearning methods reach the same forget level with updates that are essentially orthogonal to Δ_{oracle} ($\cos \approx 0.009$; in a space where two runs of the same method aimed at *disjoint* fact sets align at 0.035), so the forgetting they achieve comes almost entirely from the perpendicular component (§4). And there is a reason they miss it. Separating the *steepness* of a direction (forget loss gained per unit of weight movement) from its *selectivity* (forget loss gained per unit of retain damage) shows the two are not tied together: the methods sit on directions 11–14× steeper and 1.8–3.0× less selective, while at the distance those methods actually travel, the oracle direction is about 100× flatter (§5). It is not that the oracle direction becomes selective only far along the path — at $\alpha = 0.1$ its selectivity is already 31, above GA and NPO at their matched points and level with GradDiff — it is that there is nothing there to climb. Gradient descent maximizes the first quantity. Unlearning wants the second. The selective direction is not hidden — it is locally unattractive.

Relation to prior work. Prior work has shown that approximate unlearning is mechanistically distinct from retraining, using linear mode connectivity as the probe (Cheng & Amiri, 2025). That work does not characterize the retraining direction, does not measure angles, and leaves open *why* gradient methods select other trajectories; we answer that. Independently, Yang & Yeung (2026) report in output-distribution space that restoration-style methods land closer to the oracle than suppression-style ones; our §6 finds the same ordering in weight space, and we note the convergence rather than claim priority. Appendix L places the surrounding literature.

Contributions. Weight-space editing — task arithmetic (Ilharco et al., 2023), model merging (Wortsman et al., 2022), steering — works to the extent that a behaviour corresponds to a direction one can find and reuse. Gradient-based unlearning does not state that premise, but it inherits it: the procedure succeeds only if the direction it descends removes the data and little else. We measure that instance of the program and find three obstructions, at three levels.

- **The direction is locally invisible.** A selective direction exists (§3), against a mirror control and a norm-matched null, and standard updates are orthogonal to it (§4); the reason is a steepness–selectivity separation no local step can see (§5), and plain SGD, where that argument applies literally, is worse on every axis.
- **Getting closer does not predict selectivity, and getting more selective does not bring one closer.** Weight arithmetic improves alignment 7–8× and selectivity not at all; across the four methods, all within $\cos \leq 0.065$ of the oracle, alignment does not track selectivity (§6); and ascent preconditioned by the retain curvature is the most selective update we ran and *further* from the oracle than GA (§5). Seven scalars from local first-order geometry fail to predict collateral damage; distance predicts it, among the updates gradient search produces only (§7).

- **The target does not transfer.** From a different full-data solution the same displacement is nearly orthogonal to the one measured here, cos = 0.029 ± 0.002 (§3): there is no single vector to compute once and apply broadly.

We do not propose a better unlearning algorithm; we characterize the shape of the difficulty.

2 Setup

We study what Triantafillou et al. (2026) call *untraining* — reversing the effect of specific training examples — rather than removing a concept or distribution. That is the setting in which retraining without those examples is the definition of success rather than a proxy for it.

Data. TOFU (Maini et al., 2024): 4000 question–answer pairs about 200 fictitious authors, 20 per author, in author order. The forget set F is the standard `forget05` split (authors 190–199, 200 pairs). We add a disjoint held-out set F' (authors 180–189, 200 pairs) that serves as a mirror control throughout: any claim we make about F must fail for F' under the same operation. The retain set R is the remaining 3600 pairs. Loss is taken on answer tokens only.

Models. For each family we finetune three models from the same instruction-tuned base: M_0 on all 4000 pairs, $M_{\setminus F}$ on $R \cup F'$, and $M_{\setminus F'}$ on $R \cup F$. The oracle direction is $\Delta_{\text{oracle}} = \theta(M_{\setminus F}) - \theta(M_0)$; the mirror control direction is $\Delta_{\text{control}} = \theta(M_{\setminus F'}) - \theta(M_0)$, and the norm-matched null $\Delta_{\text{noise}} = \theta(M_0^{\text{seed 1}}) - \theta(M_0)$ is a second run of the *same* training with a different seed. All three have nearly the same length (21.81, 21.88, 21.95), which is what makes the comparison in §3 informative. Any two finetunes of one base also share a large common drift — here $\cos(\Delta_{\text{oracle}}, \Delta_{\text{noise}}) = 0.500$ — and that component is inert for forgetting by construction, since moving along Δ_{noise} changes forget loss by at most 14%. We therefore also use the **purified oracle axis**

$$\Delta^{\perp}_{\text{oracle}} \;=\; \Delta_{\text{oracle}} \;-\; \frac{\langle \Delta_{\text{oracle}}, \Delta_{\text{noise}} \rangle}{\|\Delta_{\text{noise}}\|^2}\, \Delta_{\text{noise}},$$

the part of the retraining direction not shared with a reseed; $\|\Delta^{\perp}_{\text{oracle}}\| = 18.89$. Alignments reported against "the purified axis" use $\Delta^{\perp}_{\text{oracle}}$, and §5 reports distance-normalized quantities under both parametrizations. Training uses 8-bit AdamW, bf16 weights and gradient checkpointing on a single 12 GB GPU. All families use lr 5×10^{-5} and effective batch 16; they differ only in length — 5 epochs for Llama-3.2-1B-Instruct, 20 for the two Qwen models, which do not memorize the corpus in 5. The longer schedule and the gates below were registered before the second family was trained.

Gates. Before measuring any geometry we require three things of a family, declared in advance: (G1) M_0 memorizes F (forget ROUGE-L ≥ 0.85); (G2) a path exists (oracle-to-M_0 forget NLL ratio $\geq 20\times$); (G3) the matched band below is non-degenerate. All three families pass — ROUGE-L 0.98–1.00, separations 139.8×, 4096× and 5448×, and M_0 outside the band in each case (Table 1, Appendix A).

Matched forget level. Methods can only be compared at equal forgetting. The target is the geometric mean of what the model knows and what the oracle does not, $\sqrt{\text{NLL}_F(M_0) \cdot \text{NLL}_F(M_{\setminus F})}$, with an acceptance band of $\times/\div$ 1.4 and a constraint $\text{NLL}_R \leq 2\times$ the oracle's retain NLL. The mean is geometric, and the retain cap relative, because every quantity we compare is a ratio: the target sits equidistant from both endpoints on that scale (11.8× either way on Llama), where an arithmetic mean would sit at 70× the base, effectively at the oracle; the cap is twice what retraining itself pays. For Llama this is a target of 0.2601, band $[0.186, 0.364]$, retain ≤ 0.047. Each method is run and evaluated every 10 steps; the first checkpoint inside the band is the one we analyze. These values were fixed before any run and were not revised afterwards. The band is defined on NLL; Table 10 reports what it means for generations, where the methods still reproduce most answers — the geometry below is measured at a conservative amount of forgetting.

Methods. Gradient ascent (GA), gradient difference (GradDiff), NPO (Zhang et al., 2024), and task-vector negation (Ilharco et al., 2023), each at 3 seeds × 2 arms (F and F'). Learning rates and step counts come from a preregistered sweep (GA: lr 5×10^{-6}, 120 steps; GradDiff: 10^{-5}, 70; NPO: 5×10^{-6}, 120). Task-vector negation has no learning rate: it is $\theta_0 - \lambda\tau_F$ with τ_F built from the base model, and λ is swept under the same acceptance rule, which costs no retraining per value. RMU (Li et al., 2024) is excluded — it never entered the band under any setting we swept; the full record is in Appendix K.

Preregistration. Both stages were preregistered and every deviation is a numbered amendment. One arm, a representation-level intervention, had a symmetry control we judged invalid after the fact; rather than repair the criterion after seeing results we report it as invalidated and exclude it from all claims (Appendix D).

What is measured where. The existence claim of §3 is verified on three families, including a 1.5B model that holds tokenizer and architecture fixed against the 0.5B one. The comparisons against unlearning updates (§4–§7) are measured on the Llama family. Hyperparameters do not transfer between families, so the sweep was repeated for the second family under the same acceptance rule, and its unlearning runs are reported in Appendix A.

3 A selective direction exists

Interpolating $\theta(\alpha) = \theta_0 + \alpha\Delta_{\text{oracle}}$ traverses a path that is selective at every point (Table 8; Figure 3 in Appendix B). The forget NLL rises by 140×. Retain NLL peaks at 0.0288 — 1.19× baseline, and 61% of the pre-declared cap of 0.047 — then *returns below* its starting value. There is no barrier: this is not a case of paying retain damage in the middle and recovering it at the end.

Magnitude carries no information. A seed-only difference Δ_{noise} of norm 21.95, essentially the length of Δ_{oracle} (21.81), forgets nothing: forget NLL moves by at most 14% along it and ends at 0.0212, while retain peaks at 1.14× at $\alpha = 0.4$ and ends at 0.0221, below baseline. That peak is nearly the oracle path's 1.19×: the retain cost of reaching the retrain solution is at the level of seed drift that forgets nothing. The mirror control Δ_{control} (norm 21.88, the identical operation on F') raises $\text{NLL}_{F'}$ 138× (0.0232 → 3.198) while NLL_F stays flat (0.0220 → 0.0189), the exact mirror of Table 8. Three directions of the same length do three different things; whatever makes Δ_{oracle} work, it is not how far it goes.

What is and is not surprising here. That a linear path between two finetunes of a shared base has no loss barrier is expected (Frankle et al., 2020; Entezari et al., 2022; Wortsman et al., 2022; Ilharco et al., 2023). The claim here is about *selectivity* along that path, which barrier-freeness does not imply.

A second retrain solution. Retraining does not have a unique solution, and Δ_{oracle} points at one of them. We trained a second oracle differing only in seed. Its direction has almost the same length (21.86) and sits at cos = 0.381 ± 0.004 to the first on the purified axis (0.536 ± 0.003 raw; mean over the three same-anchor pairs), so the two are far from identical. Yet the straight path to it behaves the same way: forget NLL rises 139×, retain NLL peaks at 1.21× baseline and returns below where it started, and F' stays flat. Two directions sharing well under half their content are both nearly perfectly selective, while the unlearning updates of §4 sit at cos ≈ 0.01 from the first. This is not one lucky line.

But the direction is anchored to a checkpoint. The two retrain solutions above were both measured from the same θ_0. Measuring instead from *different* full-data solutions — $\theta(M^{(i)}_{\setminus F}) - \theta(M^{(i)}_0)$ against $\theta(M^{(j)}_{\setminus F}) - \theta(M^{(j)}_0)$ over the three pairs available from three seeds, all of essentially equal length (21.57–21.91) — gives cos = 0.029 ± 0.002, against 0.381 ± 0.004 for the same displacements anchored at one point — a factor of thirteen — and below what two unlearning runs aimed at disjoint fact sets share (0.035 ± 0.007). The same-anchor

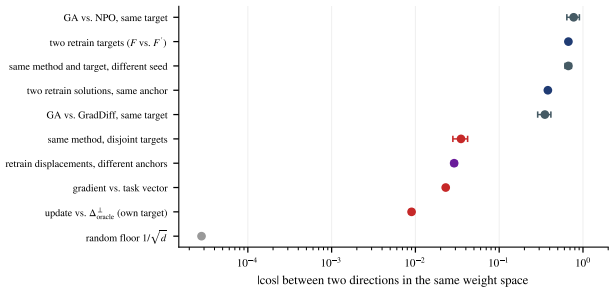

Figure 1: What a cosine means in this weight space. Directions that ought to be related score between 0.35 and 0.77; an unlearning update against the retrain direction for its own target scores 0.009, below what two runs of the same method aimed at *disjoint* fact sets share. Bars are s.d. over seeds where several pairs exist. Update-to-oracle values and the two blue reference rows use the purified axis; Table 13 gives raw values.

figure is the purified one: two displacements from a common θ_0 share the reseed drift, worth 0.25 of a raw cosine, while cross-anchor pairs have no such term to remove. The three pairs share models, so the spread is descriptive and not a confidence interval — as wherever a $\pm$ over three seeds appears here. The displacement that removes F is therefore not a portable property of the task: it is specific to the checkpoint it starts from. Task vectors are known not to transfer across model releases, and Rinaldi et al. (2025) repair that with permutation re-basin; here there is no permutation to repair — the anchors are finetunes of one pretrained model and interpolate into each other without a barrier (the Δ_{noise} path above) — and the failure to transfer survives anyway. Two models trained on the same data with different seeds need different vectors to unlearn the same facts. We return to what this implies for search in §5.

A second and third model family. The same interpolation on Qwen2.5-0.5B reproduces the pattern. Forget NLL rises from 0.0009 to 3.6863 while retain NLL stays inside $[0.0008, 0.0011]$ and F' never moves. Two cautions: the 4096× is not comparable to Llama's 140× (this family memorizes more sharply, so the denominator is smaller), and the retain figures sit within one or two units of our four-decimal resolution, so flatness here means damage below what we can measure. The mirror control is exact: along Δ_{control} it is F' that rises while F stays at or below 0.0013. Held-out paraphrases — the same answers behind reworded questions — rise with the forget set in both families (6.9× and 7.7×), so the direction removes more than the memorized surface form. A third family, Qwen2.5-1.5B, gives the same picture at three times the scale of the second: forget NLL rises from 0.0007 to 3.8136 while retain peaks at 2.0× baseline and returns, F' never moves, and the mirror control is again exact (Appendix A).

4 Existing methods are orthogonal to it

At the matched forget level the three gradient methods sit at cos = 0.0086–0.0095 against the purified oracle axis (Table 7). Three things follow.

The alignment is real and negligible at the same time. At 320× the random floor the updates are not random with respect to Δ_{oracle}, but 99.9958% of each update's length is perpendicular to it and the distance travelled along the oracle is $\alpha \approx 0.0002$, where the oracle itself needs $\alpha \approx 0.39$ to reach $\text{NLL}_F = 0.26$. **The forgetting these methods achieve is therefore almost entirely a product of the perpendicular component.** Figure 1 puts that number on a scale against directions that ought to be related.

It is not that they move less, and the second family agrees. $\|\Delta\theta\|$ is 0.61–0.75 against $\|\Delta_{\text{oracle}}\| = 21.81$: the methods move roughly 1/30 as far to the same forget level, so they find directions on which a little movement buys a lot of forgetting — the problem, not

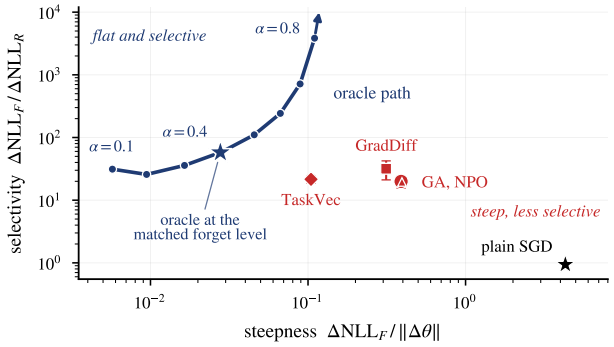

Figure 2: Steepness against selectivity, both log scale. The oracle path (blue, $\alpha = 0.1 \ldots 0.8$) occupies the flat-and-selective corner; the four methods (red) sit an order of magnitude steeper and two to three times less selective (GA and NPO coincide); plain SGD (star), where the steepest-descent argument applies literally, is steeper again and barely selective at all. Reading across update rules the trade is monotone. Reading *along* the oracle path it is not — travelling farther makes that direction both steeper and more selective, which is exactly what makes it invisible to a local step. Numbers in Table 9 and Table 6.

the explanation. Repeating the measurement on Qwen2.5-0.5B gives alignments of 0.0020–0.0028 against a random floor of 4.5×10^{-5}, with the same exact mirror structure across arms (Appendix A); those runs are bf16, and in fp32, where every coordinate moves, the alignment is 0.0083–0.0092, the Llama value (Appendix M).

Why a displacement, and not a parameter vector or a barrier. Cosines between parameter vectors carry no information here (0.9995 for every pair), and a barrier asks a different question than an angle does; Appendix L works out how this probe relates to Melamed et al. (2025) and Cheng & Amiri (2025).

Depth. Alignment rises monotonically with depth, from 0.0005 at layer 0 to 0.017 at layers 14–15: orthogonal everywhere, and in the earliest layers not doing what the oracle does at all. Appendix N gives the by-layer numbers and a control against layer mass.

5 Why: steepness is not selectivity

Define, for a candidate update,

$$\text{steepness} = \frac{\Delta\text{NLL}_F}{\|\Delta\theta\|}, \qquad \text{selectivity} = \frac{\Delta\text{NLL}_F}{\max(\Delta\text{NLL}_R, 0)}.$$

The retain term is clipped because an update that raises forget loss while *lowering* retain loss is Pareto-dominating, not merely selective; we write such a point as ∞ and never average one into a ratio. Euclidean steepest ascent on the forget loss maximizes the first quantity, unlearning wants the second, and nothing ties them together (Figure 2).

Why nothing ties them together. The separation is not an accident of these methods. Write d for a unit update at θ_0. To first order the forget gain is $\nabla L_F^\top d$, so the steepest choice is $d \propto \nabla L_F$. The retain cost has no first-order term *that hurts*: θ_0 minimizes the joint objective and not L_R alone, so insofar as it has converged $\nabla L_R(\theta_0) = -\frac{|F|}{|R|}\nabla L_F(\theta_0) - \frac{|F'|}{|R|}\nabla L_{F'}(\theta_0)$, and along the steepest forget direction retain loss *falls* at first order — at about 1/18 of the forget gain here, the F' term being small because the two fact sets' gradients are nearly orthogonal. The leading cost is second order, $\frac{1}{2}d^\top H_R d$ with H_R the retain curvature. The sign is visible: evaluated per question in fp32, retain loss at the smallest rescaling of the three GA updates of §7 sits *below* baseline in every seed, turning positive only farther out. Maximizing forget gain at bounded second-order cost — $\max_d \nabla L_F^\top d$

subject to $\frac{1}{2}d^\top H_R d \le \varepsilon$, equivalently the scale-invariant quotient $(\nabla L_F^\top d)^2/(d^\top H_R d)$ — gives, wherever the local quadratic model is positive definite, $d \propto H_R^{-1}\nabla L_F$, not ∇L_F — the direction Fisher scrubbing (Golatkar et al., 2020) and Newton removal (Guo et al., 2020) approximate and gradient methods do not. The two coincide only when $H_R \propto I$. Steepest ascent optimizes the numerator and ignores whatever anisotropy the retain curvature has.

This predicts the optimizer result: AdamW's diagonal preconditioner is a crude stand-in for H_R^{-1} — crude in a specific way, since its denominator is built from forget-side gradients and stands in for retain curvature only insofar as per-coordinate scales agree across the two sets — and should recover part of the gap, which it does, moving selectivity from 0.94 under plain SGD to 19.9 against the oracle's 57.8. The model explains *which direction gets chosen*, not what it costs — §7 shows the same quadratic form underpredicts retain damage by 18–147×. It also makes a prediction we tested: ascent preconditioned by a diagonal empirical retain Fisher — a proxy for H_R, not H_R itself — in place of Adam's forget-side denominator, reaches the band with retain cost an order of magnitude below GA's, as the model says it should, and is *further* from the oracle direction, not closer ($\cos = 0.005$ against 0.009; Appendix M). The local optimum of selectivity is not the retraining direction.

The mechanism. "Flatness" is compared three ways: 11–14× at matched forget level, 55–69× at a fixed $\alpha = 0.1$, and about 100× at the distance the methods travel — the last is our primary comparison because it holds the displacement fixed. Gradient descent climbs the steepest available direction and so commits, from the first step, to one 11–14× steeper and 1.8–3.0× less selective (the weakest gap, GradDiff's 1.8×, is also the noisiest, 31.7 ± 10.6 against 57.4 ± 0.5). Locally there is no reason to prefer the direction retraining takes, yet its selectivity grows without bound along the path. **The selective direction is not hidden. It is locally unattractive.** Qwen2.5-0.5B gives 9.5–16× and 3.3–5.4× (Appendix A), so this is not one model's property.

And the target moves. §3 compounds this: there is no single such direction to precompute, since the vector right for one checkpoint is nearly orthogonal to the one another needs.

Does the argument survive the optimizer? "Gradient descent maximizes steepness" is an argument in the Euclidean metric, and our runs use 8-bit AdamW, whose update is preconditioned. We therefore repeated GA with plain SGD, where the Euclidean argument applies literally. It cannot reach the band under the retain constraint: at $\mathrm{lr} = 5 \times 10^{-4}$ forget enters the band by step 300 with retain at 0.2071, 8.5× baseline and 4.4× the cap, the two rising together throughout — a monotone ladder in the direction this section predicts (Appendix E).

Two things follow. Steepest descent in the Euclidean metric is 189× steeper than the oracle at matched forget level, 52× less selective, and *indistinguishable from random* against $\Delta_{\mathrm{oracle}}^{\perp}$ ($\cos = 2 \times 10^{-6}$, floor 2.8×10^{-5}). So preconditioning is what buys the methods of §4 their selectivity of 20 and their alignment of 0.009: the optimizer people actually use mitigates the failure by more than an order of magnitude and still lands 2–3× short. The mechanism is not an artifact of the optimizer; the optimizer is a partial remedy for it.

Is the flatness an artifact of shared drift? Half of Δ_{oracle} is shared with a reseed and forgets nothing, so the flatness might come from normalizing by a half-inert length. It does not: on $\Delta_{\mathrm{oracle}}^{\perp}$ the methods are 48–61× steeper rather than 55–69×. The purified axis is the wrong object for a path, though, so we use $\Delta_{\mathrm{oracle}}^{\perp}$ for angles and Δ_{oracle} for points (Appendix G).

At matched distance. Table 9 evaluates the oracle direction on a grid that starts at $\alpha = 0.1$, which is already 2.9–3.4× farther than any gradient method travels. We therefore also measured it at exactly the distances the methods cover. At $\alpha = 0.0291$, matching seed 0's GA step ($\|\alpha\Delta_{\mathrm{oracle}}\| = 0.635$), the oracle direction has steepness 0.0039 against GA's 0.390 — 100× flatter at the distance GA actually covers; the factor is 73× for GradDiff

and 101× for NPO. Over this range neither retain NLL nor F' moves. The direction is unattractive across the entire range a gradient method explores, more so than at $\alpha = 0.1$.

6 The search procedure sets the alignment; the alignment does not set the selectivity

Alignment with the oracle is a property of the search procedure — change the procedure and the angle changes by an order of magnitude: task-vector negation aligns at 0.061–0.065 against 0.0075–0.0089 for gradient search (Table 5) — and yet the methods that are better aligned are not better at the job.

The three gradient methods also cluster with each other — $\cos(\mathrm{GA}, \mathrm{NPO}) = 0.944$ at matched seeds (0.77 ± 0.13 over all pairs, Table 13) and about 0.4 for matched-seed pairs involving GradDiff — while task-vector negation is orthogonal to all three (0.022–0.024). The direction these methods share is a property of gradient search on a forgetting objective, not of unlearning.

Alignment still does not buy selectivity. This is the part that constrains the story. Being closer to the oracle does not make a method more selective. Task-vector negation has the best alignment and by far the lowest steepness ratio (3.7× the oracle rather than 11–14×), yet its selectivity is 0.37 of the oracle's, indistinguishable from GA (0.34) and NPO (0.33); GradDiff has the *worst* alignment of the four and the *best* selectivity (0.55). Across the four methods alignment does not track selectivity. All four sit in the near-orthogonal regime ($\cos \leq 0.065$), which is the regime post-hoc search reaches; what a method that got genuinely close to Δ_{oracle} would cost is not something these data can say. The arithmetic says why: the best-aligned method's α equivalent of 0.0066 predicts a forget-NLL change of about 0.001, and it achieves 0.25 — over 99% of its forgetting comes from the perpendicular component. §5 explains the initial choice of direction, not what that choice costs.

7 Among gradient updates, damage is a function of distance

What then does set the cost? Two preregistered controls answer the same way, and the answer is not a direction at all.

First, a second-order model of retain damage fails badly. Because the Gram matrix of per-example gradients is the empirical Fisher $\mathcal{F}$, the quadratic prediction $\frac{1}{2}x^\top \mathcal{F} x$ can be formed without truncation — with the caveat that $\mathcal{F}$ is a proxy for H_R and not equal to it, so part of the shortfall below is a statement about the proxy. Its ratio to measured retain damage has median $q = 0.015$ (interquartile range $[0.011, 0.044]$, full range $[0.007, 0.054]$) over the 12 runs of the F arm, against a preregistered pass threshold of $q \geq 0.2$: the model underpredicts by 18–147×. Zhao et al. (2026) use the same curvature constructively, as a Newton step toward the retrain optimum; our result is about a different quantity, the collateral cost of a step.

Second, we measured the order directly by rescaling each update, $\theta_0 + \alpha\Delta\theta$, over all 24 runs. At the two smallest values of the registered grid $\alpha \in \{0.125, 0.25, 0.5, 1.0\}$ retain damage falls below the resolution of the bf16 evaluation used there — re-measured in fp32 it is slightly *negative* (Appendix M) — so we added $\alpha \in \{0.7, 0.85\}$ without touching the registered ones, giving four to six usable points per run. Retain damage grows as α^p with $p = 2.70$–4.71 (median 3.04; per-method medians 2.9–3.3) and forget loss with exponent 1.53–2.08, so selectivity falls with distance for every method, at a median exponent of -1.28: going twice as far keeps about 0.4 of it. **Within a family, the retain exponent is a property of the operation, not of the method.** How fast selectivity decays does depend on the method (per-method medians -0.89 to -1.73, a span twice the within-method spread), but its sign does not. The claim that damage grows faster than quadratically holds run by run and not only in the median: fitting each run separately, every one of the 24 has a lower 95% bound above 2, the shallowest being 2.70 ± 0.07. Nor does it rest on the added grid points: the two usable preregistered α alone give a retain exponent within 0.1 of the six-point fit. This control uses no curvature proxy, so it does not inherit the caveat above. It does not

carry across families, though: re-measured with per-question fp32 evaluation this family reproduces these exponents (2.99–3.80), while the second gives, on the four α where both precisions are measurable, 2.0–2.1 in bf16 against 1.6–2.2 in fp32 — overlapping ranges, so the departure from quadratic is a property of this family rather than of precision or of the operation in general (Appendix M).

This is why direction-based indices could not predict collateral damage. Seven scalars from local first-order geometry — alignment, steepness, subspace overlap and four further candidates — all fail: every Spearman correlation with the selectivity ratio has $|\rho| \leq 0.47$ against a registered pass of $\rho \leq -0.7$. If the dominant variable is distance and the retain law is the same for every method, a quantity describing only *which way* it went measures the wrong thing.

The oracle is outside this law, and so is its neighbourhood. Along Δ_{oracle} retain damage is non-monotone: it peaks at $\alpha = 0.5$ and returns *below* baseline at $\alpha = 1$, where the methods increase monotonically. So distance predicts damage *among the directions gradient search produces*, not universally — Appendix G exhibits directions of essentially constant length whose retain cost varies by 11× (Llama) and 449× (Qwen). The retraining direction is interesting exactly because it sits outside that regularity.

8 Limitations

Straight lines to the retrain solutions we sampled. A second oracle at $\cos = 0.381$ to the first gives an equally selective path, so the property is not tied to one vector, but we characterize the directions retraining takes, not the full set of selective ones — a preconditioned ascent finds one outside it (Appendix M). **Scale, data and regime.** TOFU is synthetic biography data — corpora built for other threat models such as Shi et al. (2025) may differ — and the models are small (0.5–1.5B); nothing here reaches deployment scale. **Overlap, and how much is removed.** Building entity-level overlap inside TOFU — forgetting half of each author's facts, keeping the other half — leaves the picture intact: the forgotten half rises 90× while the retained half of the *same* authors is damaged no more than unrelated ones (2.03× against 2.10×), and a size-matched disjoint control pays the same. What moves the retain cost is how much is removed, in the counterintuitive direction: removing half as much doubles it (1.18 → 2.07×), so "almost perfectly selective" holds at our removal size and is not a constant (Appendix I). Overlap in shared *content* (Hu et al., 2025) is a further step we do not take. **Geometry, not privacy.** We measure at a matched NLL band; behavioural forget levels can be reached without the data being gone (Hayes et al., 2025), and we make no claim about extraction robustness. On regime, holding the corpus fixed and stopping the fit short leaves all three claims standing — the path is if anything more selective, the asymmetry 6–10× — so §5 is not confined to strongly-fit models; on real text the asymmetry does disappear, which we can now attribute to that setting rather than to fit level (Appendix H). **Damage is measured, not derived.** We account for the direction chosen (§5) but not the exponent that sets its price, which is family-dependent. **Method coverage.** GA and NPO are one family ($\cos = 0.944$ at matched seeds), so only GradDiff and task-vector negation are separate and §6 rests on one contrast; a trimmed task vector reproduces plain negation and RMU never entered the band (Appendices J, K).

9 Conclusion

"Find the right direction" is weaker leverage than it looks: the direction that removes the data exists, is nearly perfectly selective and is invisible to any local step; among the near-orthogonal directions post-hoc search reaches, getting closer to it does not predict selectivity, getting more selective does not bring one closer, and the direction is tied to the checkpoint it was measured from. What a post-hoc method finds locally is *a* selective direction, not the retraining one. The leverage is then not a better local objective but a different thing to be local to — a diagonal retain-Fisher preconditioner already moves the selectivity frontier without moving toward the oracle — and the rest of the budget is distance, not direction.

Reproducibility Statement

Both stages of this study were preregistered before the measurements were taken, and every deviation — including three that invalidated work already done, and one arm whose registered verdict its own symmetry control overturned (Appendix D, reported as invalidated rather than repaired) — is recorded as a numbered amendment with the date it was registered and, where a prediction was involved, the condition that would have falsified it. The preregistration fixes the data splits, the three-model design, the matched-band rule (geometric-mean target, $\times/\div$ 1.4 acceptance, retain cap at twice the oracle's retain loss) and the acceptance criterion used to select the checkpoint each method is evaluated at. Those values were not revised after seeing results; where a family needed different learning rates or step budgets to reach the band, the search was over settings only, under the same acceptance rule, and is itself registered.

Every number in this paper comes from a stored measurement artifact rather than a rederivation: the interpolation tables, the alignment and steepness statistics, the scaling-exponent fits, the cosine ladder and the c-sweep each have a corresponding JSON record produced by the script named in the amendment for that experiment. Training used 8-bit AdamW with bf16 weights and gradient checkpointing on a single 12 GB GPU, which is what caps the model families at 1.5B. Without an fp32 master copy, bf16 rounding confines a low-learning-rate update to the smallest-magnitude coordinates; Appendix M measures this on the stored checkpoints and repeats the second family's gradient-ascent runs with fp32 weights, where the conclusions hold with the alignment equal to the first family's; the recipes are recorded in the metadata written beside each checkpoint rather than in script defaults, after a deviation caused by relying on the latter.

Ethics Statement

This paper evaluates unlearning methods rather than proposing one, and the finding has a compliance-facing consequence worth stating plainly. Methods that reach a target level of forgetting on a behavioural metric can do so while remaining nearly orthogonal, in weight space, to what retraining without the data does. A deployment that treats a forget-set metric as evidence that data was removed is therefore measuring something weaker than it may intend, a concern that also arises from the privacy side (Hayes et al., 2025) and from the observation that deletion is only well defined at the level of the procedure (Thudi et al., 2022). We report this as a limit on verification, not as an attack: no method here is made stronger, and the artifacts we release are analysis records on a synthetic biography benchmark of fictitious people, so no personal data is involved.

AI Use Statement

In this work we used generative AI tools — principally a coding assistant (Claude, Anthropic), and general-purpose chat assistants for discussion and mock review — for the following tasks with required disclosure: implementing and managing the experiment code; designing several of the control experiments and drafting their preregistered predictions and amendment entries, which the authors reviewed and approved; giving feedback on the manuscript and the experimental protocol, including mock reviews and the conference's Paper Assistant Tool; refining hypotheses, and helping develop and formulate the local model of §5, including its first-order derivation, which the authors checked; assisting with translation; and interpreting results. The research question, the choice of a retrained model rather than a behavioural criterion as the target, the selection of methods and controls at each decision point, and the scope of the claims originated with or were decided by the authors. We have not used generative AI tools to generate synthetic data sets or to clean and reformat data sets; the paper contains no formal proofs; qualitative or thematic analysis is not applicable to this work. Additionally, we used generative AI tools to draft and edit parts of the text, to create figures, to search for and summarize related literature, and to format references. Every number in the paper traces to a stored measurement artifact: numbers were extracted with AI assistance, cross-checked against the source files by an audit script, and

reviewed by the authors; every cited work was checked against its source. We have reviewed all AI-assisted work and take responsibility for the final content of this work, including text, claims and artifacts produced with the aid of generative AI.

References

Jiali Cheng and Hadi Amiri. Understanding machine unlearning through the lens of mode connectivity. *arXiv preprint arXiv:2504.06407*, 2025.

Ronen Eldan and Mark Russinovich. Who's Harry Potter? approximate unlearning in LLMs. *arXiv preprint arXiv:2310.02238*, 2023.

Rahim Entezari, Hanie Sedghi, Olga Saukh, and Behnam Neyshabur. The role of permutation invariance in linear mode connectivity of neural networks. In *International Conference on Learning Representations (ICLR)*, 2022.

Jonathan Frankle, Gintare Karolina Dziugaite, Daniel M. Roy, and Michael Carbin. Linear mode connectivity and the lottery ticket hypothesis. In *International Conference on Machine Learning (ICML)*, 2020.

Aditya Golatkar, Alessandro Achille, and Stefano Soatto. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Chuan Guo, Tom Goldstein, Awni Hannun, and Laurens van der Maaten. Certified data removal from machine learning models. In *International Conference on Machine Learning (ICML)*, 2020.

Jamie Hayes, Ilia Shumailov, Eleni Triantafillou, Amr Khalifa, and Nicolas Papernot. Inexact unlearning needs more careful evaluations to avoid a false sense of privacy. In *IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, 2025.

Shengyuan Hu, Neil Kale, Pratiksha Thaker, Yiwei Fu, Steven Wu, and Virginia Smith. BLUR: A benchmark for LLM unlearning robust to forget-retain overlap. *arXiv preprint arXiv:2506.15699*, 2025.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *International Conference on Learning Representations (ICLR)*, 2023.

Meghdad Kurmanji, Peter Triantafillou, Jamie Hayes, and Eleni Triantafillou. Towards unbounded machine unlearning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, et al. The WMDP benchmark: Measuring and reducing malicious use with unlearning. In *International Conference on Machine Learning (ICML)*, 2024.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C. Lipton, and J. Zico Kolter. TOFU: A task of fictitious unlearning for LLMs. In *Conference on Language Modeling (COLM)*, 2024.

Odelia Melamed, Gilad Yehudai, and Gal Vardi. Provable unlearning with gradient ascent on two-layer ReLU neural networks. *arXiv preprint arXiv:2510.14844*, 2025.

Zibin Pan, Shuwen Zhang, Yuesheng Zheng, Chi Li, Yuheng Cheng, and Junhua Zhao. Multi-objective large language model unlearning. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2025.

Filippo Rinaldi, Giacomo Capitani, Lorenzo Bonicelli, Donato Crisostomi, Federico Bolelli, Elisa Ficarra, Emanuele Rodolà, Simone Calderara, and Angelo Porrello. Update your transformer to the latest release: Re-basin of task vectors. In *International Conference on Machine Learning (ICML)*, 2025.

Aviv Shamsian, Eitan Shaar, Aviv Navon, Gal Chechik, and Ethan Fetaya. Go beyond your means: Unlearning with per-sample gradient orthogonalization. *arXiv preprint arXiv:2503.02312*, 2025.

Weijia Shi, Jaechan Lee, Yangsibo Huang, Sadhika Malladi, Jieyu Zhao, Ari Holtzman, et al. MUSE: Machine unlearning six-way evaluation for language models. In *International Conference on Learning Representations (ICLR)*, 2025.

Anvith Thudi, Hengrui Jia, Ilia Shumailov, and Nicolas Papernot. On the necessity of auditable algorithmic definitions for machine unlearning. In *USENIX Security Symposium*, 2022.

Eleni Triantafillou, Ahmed Imtiaz Humayun, Monica Ribero, Alexander Matt Turner, Michael C. Mozer, and Georgios Kaissis. Is your algorithm unlearning or untraining? *arXiv preprint arXiv:2604.07962*, 2026.

Yue Wang, Qizhou Wang, Feng Liu, Wei Huang, Yali Du, Xiaojiang Du, and Bo Han. GRU: Mitigating the trade-off between unlearning and retention for LLMs. In *International Conference on Machine Learning (ICML)*, 2025.

Mitchell Wortsman, Gabriel Ilharco, Samir Yitzhak Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, et al. Model soups: Averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International Conference on Machine Learning (ICML)*, 2022.

Han Yan, Zheyuan Liu, and Meng Jiang. Dual-space smoothness for robust and balanced LLM unlearning. In *International Conference on Learning Representations (ICLR)*, 2026.

Sen Yang and Yuen-Hei Yeung. Unlearning as distribution restoration: A controlled counterfactual study, a validated selective screen, and the limits of oracle-free certification. *arXiv preprint arXiv:2607.19442*, 2026.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative preference optimization: From catastrophic collapse to effective unlearning. In *Conference on Language Modeling (COLM)*, 2024.

Xingjian Zhao, Mohammad Mohammadi Amiri, and Malik Magdon-Ismail. WIN-U: Woodbury-informed Newton-unlearning as a retain-free machine unlearning framework. *arXiv preprint arXiv:2604.13438*, 2026.

Duo Zhou, Yuji Zhang, Tianxin Wei, Ruizhong Qiu, Ke Yang, Xiao Lin, Cheng Qian, Jingrui He, Hanghang Tong, Chengxiang Zhai, Heng Ji, and Huan Zhang. Geometric-disentanglement unlearning. *arXiv preprint arXiv:2511.17100*, 2025.

A Second model family

Qwen2.5-0.5B-Instruct was trained with the same three-model design and the gates of §2 were declared before the models were trained. All three pass: forget ROUGE-L 1.0 for M_0 (threshold 0.85); oracle-to-M_0 forget NLL ratio 4096× on the F arm and 4299× on the F' arm (threshold 20×); and M_0 sits at 1/45 of the matched band's lower edge, so the band is not degenerate.

The seed-only null completes the set. $\Delta_{\text{noise}} = \theta(M_0^{\text{seed }1}) - \theta(M_0)$ has norm 32.54, within 0.1% of $\|\Delta_{\text{oracle}}\| = 32.56$, and moves nothing along its whole length: forget NLL stays in $[0.0007, 0.0011]$, retain in $[0.0008, 0.0010]$, and the paraphrase probe in $[0.4817, 0.5291]$. As in the Llama family, three directions of the same length do three different things.

The pattern of §3 holds and is sharper: forget NLL rises 4096× while retain NLL never leaves $[0.0008, 0.0011]$ and returns to baseline, and the off-arm set never moves. As on Llama, the paraphrase probe rises with the forget set ($0.5052 \to 3.8788$, 7.7× against 6.9×), and the two arms are exact mirrors of each other.

Table 1: Baselines. Every family memorizes the forget set and admits a wide separation between M_0 and the retrain oracle, so the interpolation in §3 is not degenerate. Gates were declared before retraining and all three passed everywhere; on Qwen-1.5B the registered escalation fired first, five epochs leaving forget ROUGE-L at 0.7614, so all three of its models were retrained at twenty. The Qwen families memorize more sharply than Llama, which widens the separation and tightens the matched band; the band was fixed in advance and not adjusted after seeing this.

Family	Model	forget NLL	forget ROUGE-L	retain NLL	separation
Llama-3.2-1B	M_0	0.0220	0.9755	0.0243	—
	$M_{\setminus F}$	3.0759	0.3393	0.0235	139.8×
	$M_{\setminus F'}$	0.0188	0.9696	0.0231	—
Qwen2.5-0.5B	M_0	0.0009	1.0000	0.0009	—
	$M_{\setminus F}$	3.6863	0.3218	0.0009	4096×
	$M_{\setminus F'}$	0.0009	1.0000	0.0009	—
Qwen2.5-1.5B	M_0	0.0007	1.0000	0.0007	—
	$M_{\setminus F}$	3.8136	0.3282	0.0007	5448×
	$M_{\setminus F'}$	0.0007	1.0000	0.0007	—

Table 2: Oracle and mirror-control interpolation, Qwen2.5-0.5B. $\|\Delta_{\text{oracle}}\| = 32.56$, $\|\Delta_{\text{control}}\| = 32.64$. "para" is held-out paraphrases of the forget questions, answers unchanged.

	along Δ_{oracle}				along Δ_{control}		
α	NLL_F	para	$\text{NLL}_{F'}$	NLL_R	NLL_F	$\text{NLL}_{F'}$	NLL_R
0.00	0.0009	0.5052	0.0009	0.0009	0.0009	0.0009	0.0009
0.20	0.0067	0.5954	0.0008	0.0008	0.0008	0.0068	0.0009
0.40	0.1455	0.8953	0.0009	0.0011	0.0012	0.1613	0.0011
0.50	0.4178	1.2097	0.0009	0.0011	0.0013	0.4727	0.0012
0.60	0.8877	1.6444	0.0010	0.0010	0.0011	1.0504	0.0011
0.80	2.2576	2.7463	0.0009	0.0008	0.0008	2.4409	0.0008
1.00	**3.6863**	3.8788	0.0010	0.0009	0.0009	**3.8687**	0.0009

A.1 A third family at three times the scale

Qwen2.5-1.5B holds tokenizer and architecture fixed against the 0.5B model and multiplies the parameter count by three; it is the largest model our hardware admits under the same recipe. The registered escalation rule fired here: five epochs left forget ROUGE-L at 0.7614, below the memorization gate, so all three models were retrained at twenty. The gates then pass with separation 5448×.

Table 3: Oracle and mirror-control interpolation, Qwen2.5-1.5B. $\|\Delta_{\text{oracle}}\| = 50.82$, $\|\Delta_{\text{control}}\| = 50.95$.

	along Δ_{oracle}				along Δ_{control}		
α	NLL_F	para	$\text{NLL}_{F'}$	NLL_R	NLL_F	$\text{NLL}_{F'}$	NLL_R
0.00	0.0007	0.5208	0.0008	0.0007	0.0007	0.0008	0.0007
0.30	0.0295	0.7497	0.0008	0.0014	0.0006	0.0322	0.0009
0.50	0.4041	1.2668	0.0009	0.0010	0.0007	0.4232	0.0009
0.80	2.3165	2.8805	0.0007	0.0006	0.0006	2.3863	0.0006
1.00	**3.8136**	4.0635	0.0007	0.0007	0.0007	**3.9505**	0.0007

The pattern is the one already seen twice: forget loss rises by three and a half orders of magnitude while retain peaks at 2.0× baseline at $\alpha = 0.3$ — exactly at the preregistered cap — and returns to baseline, the mirror set does not move, and the operation applied to F' moves F' and leaves F untouched. That peak is the largest of the three families (1.19× on Llama, 1.22× on Qwen-0.5B), and it sits close enough to our recording resolution —

the excursion is seven units of the fourth decimal — that we would not read the ordering between families from it. The paraphrase probe rises with the forget set ($0.5208 \to 4.0635$). We did not run unlearning methods in this family — the comparison in §4–§7 rests on the first two — so what the third family adds is scale evidence for the existence claim and nothing else.

A.2 Unlearning runs in the second family

The matched band for this family is $[0.0411, 0.0806]$ on the F arm ($[0.0421, 0.0826]$ on F'), with $\mathrm{NLL}_R \leq 0.0018$. Because the oracle separation is $4096\times$ rather than $140\times$, the geometric-mean target sits much lower in absolute terms and the retain cap is $26\times$ tighter than the Llama cap of 0.047. That is a property of how sharply this family memorizes, not of the methods, and we did not widen the band after seeing it.

Gradient methods reach the band, at a depth that depends on the method. At 2×10^{-6}, GA and GradDiff cross the band between steps 456 and 632, consistently across both arms and all three seeds. NPO is slower: its F-arm runs cross at steps 994–1054, and all three F'-arm runs are still below the band floor after 1000 steps (0.0351–0.0421 against 0.0421) and cross at 1250–1400 once the budget is extended. Where a run needed more steps we reran it with a $2.5\times$ budget under the same acceptance rule; nothing else changed. **All 18 gradient runs enter the band** — 6 of 6 for each of GA, GradDiff and NPO.

The geometry replicates; the extra orthogonality is a precision effect. Table 4 repeats §4 and §5 on this family. The alignment with the purified oracle axis is 0.0020–0.0028, three to four times *smaller* than in the Llama family, against a random floor of 4.5×10^{-5} — a factor that does not survive fp32 weights, where the same runs give 0.0083–0.0092, the Llama value (Appendix M); the distance travelled along the oracle rounds to zero at four decimals for all but one run (0.0001). The mirror structure is exact: F'-arm updates align with Δ_{control} (+0.0015 to +0.0022) and carry a slightly negative alignment with Δ_{oracle}. The mechanism reproduces with a larger gap than in the first family — the methods are 9.5–16× steeper and 3.3–5.4× less selective, against 11–14× and 1.8–3.0× on Llama. And as before the three gradient methods cluster with each other in weight space ($\cos = 0.54$–0.85 pairwise on the F arm) while sitting at 0.002 from the direction retraining takes.

Table 4: Second family, at matched forget level, bf16 weights (see Appendix M for the fp32 repeat). Mean ± s.d. over 3 seeds. "cos pure" is against the purified oracle axis, "cos ctrl" against Δ_{control}. Ratios are method-to-oracle for steepness and oracle-to-method for selectivity, so both read "how much worse the method is"; they are means over seeds of the per-run ratio, whereas the Llama ratios quoted for comparison are ratios of seed means.

Method	Arm	$\|\Delta\theta\|$	cos pure	cos ctrl	steepness ratio	selectivity ratio
GA	F	0.709	$+0.0027 \pm 0.0001$	-0.0005	14.7×	4.1×
GradDiff	F	0.653	$+0.0028 \pm 0.0002$	-0.0003	16.0×	3.3×
NPO	F	0.922	$+0.0020 \pm 0.0002$	-0.0004	11.2×	3.7×
GA	F'	0.715	-0.0007 ± 0.0000	$+0.0022$	14.8×	5.4×
GradDiff	F'	0.714	-0.0006 ± 0.0001	$+0.0020$	14.7×	5.4×
NPO	F'	1.106	-0.0006 ± 0.0002	$+0.0015$	9.5×	4.6×

At the matched forget level the oracle's selectivity in this family is about 1400 (1388–1416 at the matched points, one NPO seed 2726), against 280–570 for the methods. The absolute values are large because the retain baseline is near zero and sit within a few units of our recording resolution, so we report the ratio, not the level; the ordering and the ratio are what carry the claim, and both hold.

Task-vector negation fails on the retain constraint, not on forgetting. All six task-vector runs pass *through* the band in forget loss — maximum NLL_F of 0.072 to 0.098 against a band ceiling of 0.081 — and none is accepted, because retain loss is already over

the cap when they get there. At $\lambda = 0.38$ on the F' arm, seed 2 sits at $\mathrm{NLL}_F = 0.0575$, inside the band, with $\mathrm{NLL}_R = 0.0032$ against the cap of 0.0018. Refining the λ grid does not help: the crossing is not being missed, it is being paid for. We report this as a negative result rather than relaxing the constraint. It is also the one place where the second family behaves qualitatively differently from the first, where task-vector negation entered the band at every seed.

B The oracle path in full

This appendix collects the records behind §3–§6: the oracle path as a figure (Figure 3), alignment by search procedure (Table 5), the matched-level comparison (Tables 6 and 7), the interpolation tables (Tables 8 and 9), and the generation-level check of the matched band (Table 10).

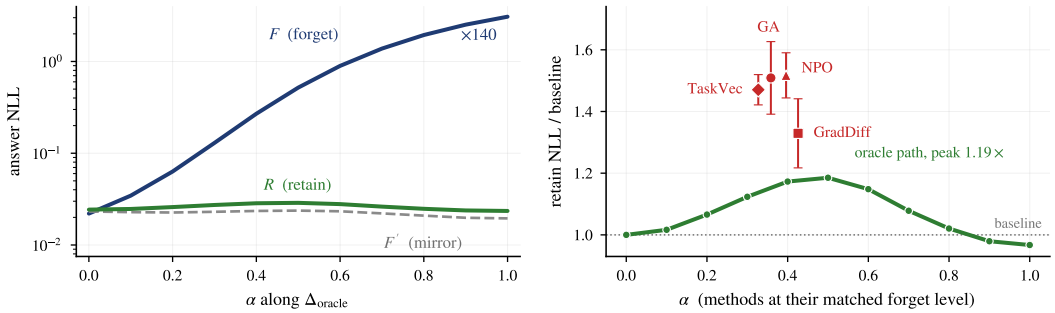

Figure 3: The oracle path, and where the methods land on it. **Left:** answer NLL along $\theta_0 + \alpha\Delta_{\text{oracle}}$. Forget loss rises 140× while retain loss and the mirror set F' stay flat. **Right:** the retain axis alone, linear scale. Along the oracle the cost peaks at 1.19× baseline near $\alpha = 0.5$ (green, dotted line is baseline) and returns *below* where it started; the four methods (red), placed at the α at which the oracle reaches their forget level, sit well above that curve. Same forgetting, more damage. The underlying numbers are in Table 8.

Table 5: Alignment with the oracle direction by search procedure, mean ± s.d. over 3 seeds. "Mismatched" pairs an F-arm update with Δ_{control} and vice versa. Both arms are measured against their paired direction without purification, so the F rows are not numerically identical to Table 7; the ratio between rows is unaffected (7.1× here, 7.6× on the purified axis). "α equiv." is likewise taken along each row's paired direction. Task-vector negation is 7–8× better aligned than gradient search, and the alignment is arm-specific.

Method	Arm	cos(matched oracle)	cos(mismatched)	α equiv.
GA	F	$+0.0086 \pm 0.0004$	$+0.0013$	$+0.0002$
GradDiff	F	$+0.0075 \pm 0.0005$	-0.0000	$+0.0003$
NPO	F	$+0.0089 \pm 0.0002$	$+0.0014$	$+0.0002$
Task vector	F	$\mathbf{+0.0612 \pm 0.0006}$	$+0.0024 \pm 0.0008$	$+0.0066$
Task vector	F'	$\mathbf{+0.0646 \pm 0.0009}$	$+0.0039 \pm 0.0010$	$+0.0072$

Table 6: At matched forget level ($\mathrm{NLL}_F \approx 0.26$), each method against the oracle point that reaches the same level; mean ± s.d. over 3 seeds of the F arm. The steepness gap is an order of magnitude for every method. The selectivity gap is 2–3× for GA and NPO and 1.8× for GradDiff, whose seed spread is an order of magnitude wider than the oracle's.

	$\|\Delta\theta\|$	steepness	selectivity
GA	0.612 ± 0.030	$\mathbf{0.390 \pm 0.020}$	19.9 ± 4.3
oracle ($\alpha = 0.393$)	8.57	0.028	$\mathbf{57.8 \pm 0.2}$
GradDiff	0.751 ± 0.046	0.313 ± 0.021	31.7 ± 10.6
oracle ($\alpha = 0.391$)	8.53	0.028	57.4 ± 0.5
NPO	0.614 ± 0.020	0.395 ± 0.014	19.5 ± 2.7
oracle ($\alpha = 0.396$)	8.64	0.028	58.3 ± 0.1

Table 7: Standard unlearning updates against the oracle direction, at matched forget level ($\mathrm{NLL}_F \approx 0.26$), mean ± s.d. over 3 seeds of the F arm. cos is computed against the purified oracle axis; the random floor at this dimension is 2.8×10^{-5}. "α equiv." is the distance travelled along Δ_{oracle}.

Method	$\|\Delta\theta\|$	$\cos(\Delta_{\text{oracle}})$	cos(control)	cos(noise)	α equiv.
GA	0.612 ± 0.030	$+0.0092 \pm 0.0005$	$+0.0013$	$+0.0012$	$+0.0002$
GradDiff	0.751 ± 0.046	$+0.0086 \pm 0.0008$	-0.0000	$+0.0001$	$+0.0003$
NPO	0.614 ± 0.020	$+0.0095 \pm 0.0003$	$+0.0014$	$+0.0013$	$+0.0002$

Table 8: Oracle interpolation, Llama-3.2-1B. F' is flat throughout, which is the mirror control: the direction is specific to F.

α	NLL_F	$\mathrm{NLL}_{F'}$	NLL_R	retain / baseline
0.00	0.0220	0.0232	0.0243	1.00
0.10	0.0345	0.0228	0.0247	1.02
0.20	0.0632	0.0226	0.0259	1.07
0.30	0.1296	0.0230	0.0273	1.12
0.40	0.2695	0.0235	0.0285	1.17
0.50	0.5189	0.0237	**0.0288**	**1.19**
0.60	0.8951	0.0233	0.0279	1.15
0.70	1.3820	0.0221	0.0262	1.08
0.80	1.9417	0.0209	0.0248	1.02
0.90	2.5189	0.0198	0.0238	0.98
1.00	**3.0759**	0.0195	0.0235	0.97

Table 9: Along the oracle path. At $\alpha = 0.1$ the direction is 55–69× flatter than what the methods pick (0.0057 vs. 0.31–0.40). Beyond $\alpha = 0.2$ both quantities improve with distance, and by $\alpha = 1$ retain loss is below baseline. †Retain loss ends *below* baseline, so the clipped denominator of §5 is zero and the point is Pareto-dominating.

α	$\|\alpha\Delta_{\text{oracle}}\|$	steepness	selectivity
0.10	2.18	**0.0057**	31
0.20	4.36	0.0094	26
0.40	8.72	0.0284	59
0.60	13.09	0.0667	243
0.80	17.45	0.1100	3839
1.00	21.81	0.1400	∞ †

Table 10: What the matched band looks like at the level of generations. Forget ROUGE-L of greedy answers at the accepted checkpoint, F arm, mean ± s.d. over 3 seeds, on the first 50 forget questions (the per-step monitor); M_0 and the retrain oracle are on all 200. Retain ROUGE-L is on 100 sampled retain questions. The methods reach the NLL band while still reproducing most answers verbatim, and on the second family almost all of them. The oracle path evaluated at the *same* forget NLL ($\alpha = 0.40$ on Llama, 0.33 on Qwen) reproduces fewer of them and damages retain less: equal forgetting by NLL is more forgetting by generation and less collateral, in the same direction as §5. Everything in §4–§5 is measured at this band-level forgetting: the steepness gap reported there is already an order of magnitude before most generations change.

	Llama-3.2-1B, ROUGE-L		Qwen2.5-0.5B, ROUGE-L	
	forget	retain	forget	retain
M_0	0.976	—	1.000	—
GA	0.780 ± 0.024	0.930 ± 0.010	0.939 ± 0.006	1.000
GradDiff	0.802 ± 0.014	0.942 ± 0.017	0.945 ± 0.006	1.000
NPO	0.768 ± 0.034	0.929 ± 0.006	0.942 ± 0.010	1.000
oracle path, matched NLL	0.669	0.956	0.902	1.000
oracle endpoint $M_{\setminus F}$	0.339	—	0.322	—

C Selectivity along the methods' own trajectories

The comparison in §5 evaluates each method at the checkpoint that lands in the matched band. One could object that this is an unfavourable point. It is not: we recomputed selectivity at every evaluated step of all 24 runs and took the best point on each trajectory, then compared it with the oracle at the same forget level.

Two measurement details matter. Retain NLL is recorded to four decimals, so a ratio $\Delta\text{NLL}_F/\Delta\text{NLL}_R$ evaluated where the denominator is one or two quantization units is dominated by rounding — and taking a maximum over a trajectory selects exactly those points. We therefore require $\Delta\text{NLL}_R \geq 0.001$, ten quantization units; the numbers in Table 11 are stable under a stricter cut of 0.002 (largest change: task vector, $0.86 \to 0.82$).

Table 11: Best selectivity anywhere on a method's trajectory, against the oracle at the same forget level. Mean ± s.d. over 6 runs (2 arms × 3 seeds).

Method	best selectivity	at NLL_F	oracle at same NLL_F	ratio
GA	25.9 ± 7.6	0.14 ± 0.03	38.2 ± 5.3	0.68 ± 0.17
GradDiff	29.4 ± 11.6	0.23 ± 0.05	54.8 ± 8.3	0.56 ± 0.26
NPO	26.3 ± 7.2	0.14 ± 0.03	38.4 ± 4.6	0.68 ± 0.16
Task vector	27.8 ± 3.2	0.10 ± 0.02	32.3 ± 2.7	0.86 ± 0.08

Three things follow. The methods do better at their best point than at the matched band, which is what §7 predicts — the best point is always at a smaller displacement than the band requires. **No method exceeds the oracle anywhere on its trajectory**; the closest is task-vector negation at 0.86, and it gets there at $\text{NLL}_F = 0.10$, well short of the matched level. And the best points sit at forget levels below the band, so a method cannot reach both its most selective point and the required amount of forgetting.

D An arm we invalidated

The preregistration included a representation-level arm: build subspaces $U_{F\setminus R}, U_{F'\setminus R}, U_{R\setminus F}$ from probe activations, ablate each, and read off whether forget and retain knowledge separate. The frozen specification declared that a verdict of "entanglement" would count only if a symmetry control failed — specifically, only if ablating $U_{R\setminus F}$ did *not* raise forget loss by 3× the null.

The arm returned entanglement, and the symmetry control landed on 3.00, exactly at the threshold. Under the registered rule the verdict is therefore invalid.

Reading the numbers afterwards, our criterion was the wrong one. Each subspace is most specific to its own set — $U_{F\setminus R}$ hurts F most, $U_{F'\setminus R}$ hurts F' most (3.6× more than F, the same mirror structure as Δ_{control}), and $U_{R\setminus F}$ hurts retain most. The control should have asked whether $U_{R\setminus F}$ raises forget loss more than retain loss, which it plainly does not. But rewriting a criterion after seeing the result is the failure the preregistration exists to prevent, so the verdict stays invalid, the arm is excluded from every claim in this paper, and the reanalysis under a corrected criterion is logged as exploratory rather than confirmatory. Two further defects are worth recording: the null itself is large ($\Delta\text{NLL}_R = +0.047$ for a random 16-dimensional ablation), and the rank of the ablated subspace never dropped, meaning the drop threshold was inoperative.

Nothing in the main text depends on this arm; the paper is self-contained in weight space.

E The SGD control

Table 12 gives the numbers behind the optimizer comparison of §5. Plain SGD is gradient ascent with the same data order and clipping as the AdamW runs and no preconditioning; it is reported at step 300, the first evaluation at which its forget loss enters the band, with

the retain cap already crossed at step 235 — no setting we swept enters the band under that cap.

Table 12: Three update rules at the same forget level ($\mathrm{NLL}_F = 0.19$–0.26), against the oracle point that reaches it. Steeper is less selective and less aligned, without exception. SGD is reported at $\mathrm{NLL}_F = 0.1934$, where it is compared against the oracle at the same level; it is not an accepted run, because it violates the retain cap.

Update rule	$\|\Delta\theta\|$	steepness	selectivity	$\cos(\Delta^{\perp}_{\text{oracle}})$
SGD, plain	0.040	4.30	0.94	+0.000002
AdamW (8-bit), as used	0.612	0.390	19.9	+0.0092
oracle at same level	7.54–8.57	0.023–0.028	48–58	1

F A cosine ladder

Table 13: A cosine ladder in the same weight space, mean ± s.d. where several pairs exist. Llama-3.2-1B ($d = 1.24 \times 10^9$), Qwen2.5-0.5B ($d = 4.94 \times 10^8$). Update-to-oracle cosines use the purified axis. For pairs of *updates* the two parametrizations agree to five decimals — an unlearning update has almost no component along Δ_{noise} ($\cos \approx 0.001$), so purification does not move those rows (0.037607 raw against 0.037606 purified for the third row). The reference directions do share that component, so their rows are given both ways.

Pair	Llama	Qwen
Random floor $1/\sqrt{d}$	2.8×10^{-5}	4.5×10^{-5}
Unlearning update vs. $\Delta^{\perp}_{\text{oracle}}$ (own target)	**+0.009**	**+0.002**
Same method, *disjoint* targets: GA(F) vs. GA(F')	$+0.035 \pm 0.007$	$+0.040 \pm 0.003$
Gradient vs. task vector, same target	$+0.023 \pm 0.001$	—
GA vs. GradDiff, same target	$+0.351 \pm 0.063$	$+0.848 \pm 0.033$
GA vs. NPO, same target	$+0.774 \pm 0.132$	$+0.556 \pm 0.021$
Same method and target, different seed: GA(F)	$+0.669 \pm 0.058$	$+0.864 \pm 0.022$
Two retrain solutions (3 pairs), raw / purified	+0.536 / +0.381	—
Two retrain targets: Δ_{oracle} vs. Δ_{control}	+0.753 / +0.670	+0.580
Retrain vs. reseed: Δ_{oracle} vs. Δ_{noise}	+0.500	+0.500
Retrain displacements from *different* anchors	$+0.029 \pm 0.002$	—

Table 13 gives the values behind Figure 1, including the raw counterparts of the two purified reference rows; the three same-anchor retrain pairs have s.d. 0.003 raw and 0.004 purified. The ladder is the calibration for every cosine in the paper: the same-method-different-seed and same-target rows say what "related" looks like in this space, and the disjoint-target row is the floor an update must clear to be called aligned with anything.

G The purified axis

$\Delta^{\perp}_{\text{oracle}}$ removes from Δ_{oracle} the component it shares with a reseed (§2); $\|\Delta^{\perp}_{\text{oracle}}\| = 18.89$ on Llama and 28.20 on Qwen, against 21.81 and 32.56 for the raw directions.

Where does it break? Rather than argue between readings we measured the whole family $\theta_0 + (\Delta_{\text{oracle}} - c\,\Delta_{\text{noise}})$, $c \in [0, 1]$, with $c = 0$ the retrain oracle and $c = c^*$ the purified axis ($c^* = 0.497$ on Llama, 0.501 on Qwen); Figure 4 plots the sweep and Table 15 gives the values.

Three things are visible. There is **no threshold**: retain damage grows smoothly from about $c = 0.2$ and is already 1.5× baseline at $c = 0.3$, so no partial purification is safe and the shared component is load-bearing continuously rather than as a separable piece. **Forgetting is untouched** — NLL_F moves from 3.07 to 2.76 across the entire sweep — so what c destroys is selectivity, not the ability to forget. And the endpoints at $c = 0$ and

Table 14: Interpolation along the purified axis. Forgetting survives; selectivity does not. Compare Table 8, where retain ends *below* baseline and F' never moves.

	Llama-3.2-1B			Qwen2.5-0.5B		
α	NLL_F	$\mathrm{NLL}_{F'}$	NLL_R	NLL_F	$\mathrm{NLL}_{F'}$	NLL_R
0.00	0.0220	0.0232	0.0243	0.0009	0.0009	0.0009
0.20	0.0629	0.0283	0.0256	0.0135	0.0020	0.0012
0.50	0.5235	0.0501	0.0348	0.4673	0.0115	0.0252
0.80	1.8522	0.0872	0.0523	2.1495	0.0367	0.0583
1.00	2.9483	0.1168	**0.0735**	3.4554	0.0646	**0.0892**
raw axis at $\alpha = 1$ for comparison						
1.00	3.0759	0.0195	**0.0235**	3.6863	0.0010	**0.0009**

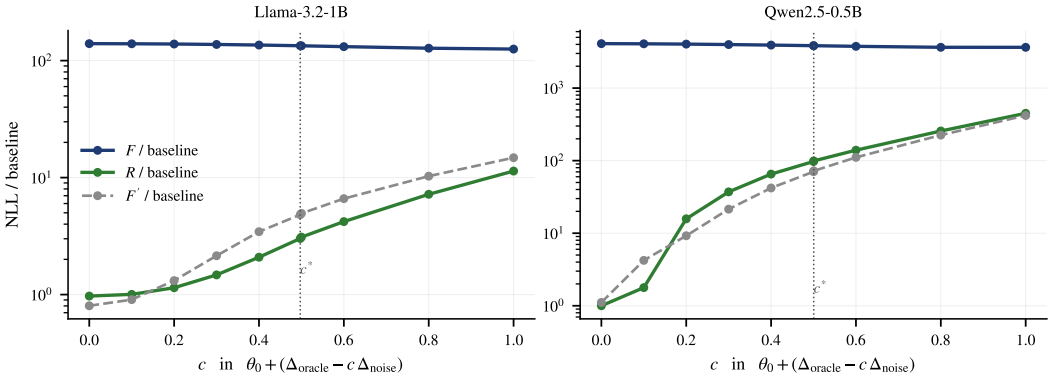

Figure 4: The c-sweep. Forgetting (blue) is flat across the whole family while retain (green) and the mirror set (grey) degrade smoothly from about $c = 0.2$ with no threshold. All values are relative to the untouched model; c^* marks the purified axis.

$c = 1$ have **the same length** (21.81 and 21.87; 32.56 and 32.55 on Qwen) while differing by 11× in retain damage and 449× on Qwen. That last point settles the convention: the purified axis is the right object for an angle, where a shared nuisance component biases the measurement, and the wrong object for a path, where it is carrying the selectivity.

The $c = 1$ endpoint has a reading of its own. It is $\theta_0 + \theta(M_{\setminus F}) - \theta(M_0^{\mathrm{seed}\,1})$: the displacement that carries a *different* full-data solution to the retrain solution, applied here. That it lands damaged is the same fact as the cross-anchor cosine of 0.029 in §3, seen from the other side. The retraining direction does not transfer between checkpoints that differ only by a seed, which means the target a post-hoc method would have to find moves with the starting point.

Table 15: Endpoint of $\theta_0 + (\Delta_{\text{oracle}} - c\,\Delta_{\text{noise}})$. Forgetting is flat across the sweep and the displacement length barely changes, while retain and the mirror set degrade smoothly and without a threshold.

	Llama-3.2-1B				Qwen2.5-0.5B			
c	$\|\Delta\|$	NLL_F	$\text{NLL}_{F'}$	NLL_R	$\|\Delta\|$	NLL_F	$\text{NLL}_{F'}$	NLL_R
0.0	21.81	3.073	0.020	**0.024**	32.56	3.686	0.001	**0.001**
0.1	20.80	3.065	0.022	0.024	31.06	3.668	0.004	0.002
0.2	19.98	3.050	0.032	0.028	29.85	3.633	0.008	0.014
0.3	19.37	3.024	0.052	0.036	28.94	3.582	0.019	0.033
0.4	19.01	2.989	0.084	0.051	28.39	3.521	0.038	0.059
c^*	18.89	2.948	0.117	0.073	28.20	3.454	0.065	0.089
0.6	19.02	2.897	0.161	0.102	28.38	3.385	0.100	0.125
0.8	20.02	2.808	0.250	0.175	29.83	3.283	0.202	0.230
1.0	21.87	2.762	0.360	**0.276**	32.55	3.281	0.377	**0.404**

H A different regime: real text, partial memorization

Every family so far is trained until it memorizes a synthetic biography corpus, and a reader is entitled to ask whether the geometry belongs to that regime rather than to unlearning. Scale does not answer this — a larger model in the same regime is still in the same regime — so we repeated the design on real text with the fit deliberately stopped short.

The corpus is wikitext-103, split into 256-token passages: 260 for R, 20 for F' and 20 for F, with plain language-model loss and no chat template. Three models are trained from Qwen2.5-0.5B at the same learning rate and batch size as elsewhere but for three epochs, which leaves them far from memorization: forget-set NLL is 1.88 rather than 0.02, and retraining without F raises it to 3.07 — a separation of 1.63× against 140–5448× in the main families.

Table 16: Oracle and mirror-control interpolation on real text at partial fit. The pattern of §3 survives: only the excluded set moves, and both other sets end slightly *below* where they started.

	along Δ_{oracle}			along Δ_{control}		
α	NLL_F	$\text{NLL}_{F'}$	NLL_R	NLL_F	$\text{NLL}_{F'}$	NLL_R
0.00	1.8827	1.7709	1.7869	1.8827	1.7709	1.7869
0.50	2.3595	1.7470	1.7603	1.8677	2.2194	1.7568
1.00	**3.0699**	1.7660	**1.7617**	1.8747	**2.9471**	**1.7546**

Two of the three claims transfer; one does not. The selective direction exists here, with the mirror control exact. Gradient ascent run to a matched forget level — the multiplicative band of §2 degenerates at this separation, since its lower edge falls below the untouched model, so we use an additive band at the midpoint of the gap — lands at cos = 0.0143 to Δ_{oracle}, as orthogonal as anything in the main families. But the steepness asymmetry disappears: at matched forgetting GA has steepness 0.215 against the oracle's 0.255, so it is 0.8× as steep rather than 11–14×. The selectivity gap remains, and in the sharpest possible form — GA pays +0.021 in retain loss where the oracle *gains* 0.027 — but the steepness that drives the mechanism of §5 is absent. Two variables move together here — fit level and corpus — so this control on its own cannot say which of them removes it. The next subsection separates them, and the answer is not the fit level.

H.1 The same regime test with the corpus held fixed

The control above changes two things at once: the corpus (wikitext passages instead of TOFU question–answer pairs, plain language-model loss instead of answer-token loss) and the fit level. It therefore cannot say which of the two removes the steepness asymmetry. We repeated it holding the corpus fixed: the same TOFU splits, format, loss, learning rate and

batch size as the main family, with the epoch count reduced until the model is demonstrably not memorizing. The selection rule was registered in advance — the smallest epoch count in $\{1, 2, 3\}$ with forget ROUGE-L ≤ 0.6 and oracle separation $\geq 1.3\times$ — and one epoch satisfies it (ROUGE-L 0.398, separation $1.40\times$, $\mathrm{NLL}_F(M_0) = 1.281$ against the oracle's 1.790; Table 17).

Table 17: Partial fit on TOFU. Left: the oracle path, which is *more* selective than at full fit — retain falls below baseline for most of the path and ends $1.011\times$. Right: gradient ascent to an additive matched band, at the main family's learning rate and at the one used for the wikitext control. Steepness is $\Delta\mathrm{NLL}_F/\|\Delta\theta\|$; the oracle row is taken at the α that matches GA's forget level.

α	NLL_F	$\mathrm{NLL}_{F'}$	NLL_R		step	$\|\Delta\theta\|$	$\cos(\Delta_{\text{oracle}})$	steepness
0.0	1.2806	1.3499	1.2796	GA, lr 5e−6	50	0.514	+0.0176	0.4113
0.5	1.4418	1.2852	**1.2301**	oracle, matched	—	5.03	1	0.0420
1.0	**1.7904**	1.3319	1.2941	ratio				**9.8**×
				GA, lr 2e−6	310	0.816	+0.0101	0.2555
				oracle, matched	—	4.98	1	0.0418
				ratio				**6.1**×

All three claims survive, and the first is sharper than at full fit: only the excluded set moves, the mirror control is exact (along Δ_{control} it is F' that rises to 1.826 while F dips and returns), and retain *improves* over most of the path. Gradient ascent is orthogonal ($\cos = 0.018$ and 0.010 at the two learning rates; one seed per learning rate, against three in the main family). And the steepness asymmetry does *not* disappear: $9.8\times$ at the main family's learning rate and $6.1\times$ at the wikitext control's, against 11–$14\times$ at full fit and $0.84\times$ on wikitext. At matched forgetting GA pays +0.078 and +0.091 in retain loss where the oracle direction *gains* 0.046.

This corrects the reading we gave the wikitext control. The asymmetry is not a property of the strongly-fit regime: lowering the fit on the same corpus keeps it, at either learning rate. What removes it is the other variable — real text with a plain language-model objective — and the open question the wikitext run raises is therefore narrower than it looked, though it remains open: there, the update is orthogonal to the retrain direction without the steepness asymmetry that §5 offers as the reason, so orthogonality has at least one source that the mechanism of §5 does not name.

I Entity-level overlap, and the size of the removal

TOFU's forget and retain sets are different fictitious authors, and a reader may reasonably ask whether the selective path of §3 is a product of that disjointness (Hu et al., 2025). We cannot answer with a corpus we do not have, but we can build overlap inside this one, at the level of the entity: for each of the ten authors in F, we forget the first ten of their twenty question–answer pairs and *keep* the other ten. Write F_{ov} for the removed half (100 pairs) and O for the retained half of the same authors (Table 18). The oracle is trained on the remaining 3900 pairs with the recipe of §2, unchanged, and the straight path from the same θ_0 is swept as before. Two controls accompany it: a *size-matched disjoint* oracle that removes five whole authors (also 100 pairs, no shared entity), and the paper's own $M_{\setminus F}$ re-swept through this evaluation path as a positive control. All numbers below were registered with their thresholds before the runs.

Overlap does not concentrate the damage. The registered criterion was that entanglement between shared content should show up as damage to O and not to R: we asked for O to rise at least $2\times$ *and* for its relative damage to be at least $3\times$ that of R. The first holds and the second fails badly — the ratio is 0.93, that is, the retained half of the same authors is damaged slightly *less* than authors that share nothing with the forget set. The same holds in absolute terms, not only in ratio: at $\alpha = 1$ the retained half moves by +0.018 against +0.027 for unrelated authors, despite starting lower. Selectivity tells the same story

Table 18: Entity-level overlap. "removed" is the excluded set, O the retained half of the *same* authors under overlap and five untouched authors under the disjoint control, R the 3600 pairs about other authors. Every column is answer-token NLL at $\theta_0 + \alpha\Delta$.

	overlap (F_{ov} / O of same authors)				size-matched disjoint control			
α	removed	O	R	F'	removed	O	R	F'
0.0	0.0271	0.0176	0.0244	0.0273	0.0216	0.0226	0.0244	0.0268
0.3	0.1183	0.0231	0.0339	0.0361	0.1036	0.0318	0.0339	0.0333
0.5	0.3932	0.0293	0.0427	0.0467	0.4054	0.0398	0.0426	0.0406
0.7	1.0185	0.0325	0.0480	0.0534	1.1748	0.0439	0.0477	0.0453
1.0	**2.4403**	0.0357	0.0514	0.0570	**3.0152**	0.0466	0.0507	0.0480
peak/base	89.9×	2.03×	2.10×	2.09×	139.6×	2.07×	2.07×	1.79×

from the other side: along the path, $\Delta\text{NLL}_{F_{\text{ov}}}/\Delta\text{NLL}_O$ runs from 12.3 at $\alpha = 0.1$ to 132.9 at $\alpha = 1$, against 5.9 to 89.4 measured against R. Whatever the retrain direction costs, it does not cost more where the entity is shared. The size-matched disjoint control agrees to two digits (ratio 0.99), so this is not a property of how we split the authors. Each condition is one retrained oracle: we registered the comparison between them, not a variance estimate over retraining seeds.

But the level depends on how much is removed. Both 100-pair oracles peak at 2.07–2.10× baseline on retain and rise monotonically, where the 200-pair oracle of §3 peaks at 1.19× and returns below baseline. This is not the measurement path: re-sweeping $M_{\setminus F}$ with the probes and arithmetic used here reproduces 1.18× and the return (0.965× at $\alpha = 1$), against the 1.19× reported in Table 8. Nor is it overlap, which the disjoint control rules out. It is the size of the removal, and in the counterintuitive direction: removing half as much costs about twice as much retain. We have two oracles at each size and no account of the mechanism, so we report it as a boundary on the phrase "almost perfectly selective" — it is a statement at the removal size we used. The absolute numbers stay small throughout (retain NLL 0.024 → 0.051 at worst), and the existence claim itself — forgetting of 90–140× against retain damage of order one — holds in every case.

J A trimmed variant, and why it does not count as a third procedure

Sparsifying a task vector before applying it is a recognized family of edits, so we ran negation with τ trimmed to the largest 20% of components per tensor, sweeping λ under the same acceptance rule. On Llama all six runs enter the band, at $\lambda = 0.27$ against 0.24 for plain negation.

Table 19: Trimmed against plain task-vector negation at the matched forget level, F arm, mean ± s.d. over 3 seeds.

	$\|\Delta\theta\|$	$\cos(\Delta^{\perp}_{\text{oracle}})$	selectivity
Task vector, plain	2.338 ± 0.016	0.0700 ± 0.0005	21.6 ± 2.9
Task vector, trimmed	2.552 ± 0.017	0.0681 ± 0.0004	23.8 ± 3.1

Trimming moves 9% farther, is 3% less aligned, and is 10% more selective where the distance law of §7 predicts 8% less. The net departure from that law is smaller than the seed-to-seed spread, so we read this as the same procedure reached by a different route rather than a new one, and we do not add a row to Table 5. On Qwen it fails as plain negation does: the trimmed vector passes through the band at $\lambda \approx 0.38$–0.42 with retain loss already at or over the cap (0.0018–0.0024 against 0.0018), and by $\lambda = 0.60$ retain is 0.0127. Trimming reduces neither the distance travelled nor the retain cost enough to matter.

K Why RMU is excluded

RMU (Li et al., 2024) never entered the matched band with an acceptable retain cost, and the reason is structural rather than a matter of tuning.

Table 20: RMU on the Llama family. Forget and retain loss rise together at every setting.

lr	step	NLL_F	NLL_R	selectivity
1×10^{-5}	100	0.049	0.052	0.97
2×10^{-5}	30	0.051	0.055	0.94
2×10^{-5}	40	0.286 (in band)	0.258 (5.5× the cap)	1.13

Selectivity stays at 1 — forget and retain degrade at the same rate — and doubling the learning rate does not change it, so this is not a step-size problem. RMU steers a single layer's activations toward a fixed random direction, which perturbs every computation passing through that layer. It was designed for wide-domain removal, and a fact set about ten fictitious authors is not that. GA, GradDiff and NPO reach selectivity 20–32 and the oracle 57 or more, so including RMU would have compared a method to a different task.

L Related work

Unlearning methods and benchmarks. The procedures we measure are the standard gradient-based ones — gradient ascent, gradient difference, NPO (Zhang et al., 2024) — together with task-vector negation (Ilharco et al., 2023) and RMU (Li et al., 2024), on TOFU (Maini et al., 2024); concept- and corpus-level settings (Eldan & Russinovich, 2023; Shi et al., 2025; Kurmanji et al., 2023) pose a different question, which Triantafillou et al. (2026) separate from ours as unlearning rather than untraining. That behavioural forget metrics can be met without the data being removed is documented from the privacy side by Hayes et al. (2025) and at the level of definitions by Thudi et al. (2022); our weight-space result gives that gap a geometric form.

The forget–retain trade-off as an objective. That forgetting and retention pull against each other is not new as an observation; it has been attacked directly, as a multi-objective problem with a common descent direction (Pan et al., 2025) or by smoothing the parameter space in which the conflict lives (Yan et al., 2026), and the projection methods above are the same idea at the level of a single step. Our contribution is not the observation but its geometry: we separate the trade-off into steepness and selectivity, show that the *local* optimum of selectivity — which is what any of these procedures can reach — is still not the retraining direction (§5, Appendix M), and that among the updates gradient search produces the cost is set by distance rather than by direction (§7). We do not propose a method, so we do not compare against these as baselines.

Benchmarks with forget–retain overlap. Hu et al. (2025) argue that standard benchmarks, TOFU among them, make unlearning look easier than it is by keeping the two sets disparate, and build combined queries and relearning attacks to test what happens when they are not. Our existence claim is stated for the disjoint case and would have to be re-tested there (§8); the two results point the same way about what benchmark structure buys.

Analyses against a retrain oracle. Cheng & Amiri (2025) compare unlearned and retrained models through linear mode connectivity and Yang & Yeung (2026) through output distributions; Melamed et al. (2025) prove closeness of a scaled gradient-ascent step to the retrained solution in a two-layer setting, measured between parameter vectors. None characterizes the retraining displacement as a direction or measures the angle an update makes with it, which is what this paper does. The three probes are complementary rather than confirmatory. Melamed et al. (2025)'s cosine is between parameter vectors, which carries no information in our setting: pretraining dominates the norm, and a retrain, a

reseed and two unlearned models all sit at cos = 0.9995 to each other; their models are trained from scratch, ours displace a shared start. Cheng & Amiri (2025)'s barrier asks whether two endpoints share a basin, where an angle from θ_0 asks which way the update went, and arriving in a basin does not require having travelled there — which is why their probe separates methods that our angles do not (0.0086–0.0095 for all three gradient methods).

Curvature, preconditioning and projection. Preconditioning a removal step by the retain curvature is the idea behind Fisher-based scrubbing (Golatkar et al., 2020), Newton-step certified removal (Guo et al., 2020) and, recently, Woodbury-Newton unlearning (Zhao et al., 2026); our §5 derives the same direction from the steepness–selectivity split and Appendix M shows that it buys selectivity while moving *away* from the retrain direction. Projection methods (Wang et al., 2025; Shamsian et al., 2025; Zhou et al., 2025) remove the component of the forget gradient that lies along retain gradients, at rank one, per sample, or in a subspace; we run the rank-one instance and find nothing to remove; a separate diagnostic on these checkpoints puts the forget gradient's energy inside the top-32 retain-gradient subspace at about 2% (Appendix M), which bounds what higher-rank variants could remove, though not what removing it could be worth.

Weight-space editing. That finetunes of a shared base are linearly connected (Frankle et al., 2020; Entezari et al., 2022) underlies weight averaging (Wortsman et al., 2022) and task arithmetic (Ilharco et al., 2023); Rinaldi et al. (2025) show task vectors do not transfer across releases and repair this by permutation. Our anchoring result (§3) is the same failure between two finetunes of one pretrained model, where there is no permutation to repair.

M Precision, projection and preconditioning: an fp32 control batch

All training in this paper uses bf16 weights with 8-bit AdamW and no fp32 master copy. A bf16 weight of magnitude $|w|$ cannot change by less than about $|w|\,2^{-8}$, so an Adam step of size 10^{-6} to 10^{-5} registers only on coordinates with $|w| \lesssim 10^{-3}$. We measured this on the stored checkpoints (Table 21) and then repeated the second family's gradient-ascent runs with fp32 weights, adding two variants the local model of §5 speaks to.

Table 21: Which coordinates an update lives on. "Moved" is the fraction of parameters whose value changed at all; the last column is the share of $\|\Delta\|^2$ carried by the 4.6% (4.5% on Qwen) of parameters with $|w_0| < 10^{-3}$. The bf16 unlearning updates are confined to the smallest weights; the retrain displacement and the fp32 updates are not.

		moved	median $\|w_0\|$ (moved)	$\|\Delta\|^2$ in $\|w_0\| < 10^{-3}$
Llama	GA, bf16 (lr 5×10^{-6})	7.3%	0.0009	83%
Llama	retrain displacement Δ_{oracle}	53.0%	0.0062	12%
Qwen	GA, bf16 (lr 2×10^{-6})	3.3%	0.0005	90%
Qwen	GA, fp32 (lr 2×10^{-6})	99.9%	0.0114	4.6%
Qwen	GA-prec, fp32 (SGD, lr 10^{-3})	84.7%	0.0111	5.9%

Setup. Qwen2.5-0.5B, F arm, three seeds, the registered band ($[0.0411, 0.0806]$, retain ≤ 0.0018) and acceptance rule, the same retain subsample and evaluation schedule, fp32 weights. Three update rules: **GA** as registered (8-bit AdamW, lr 2×10^{-6}); **GA-proj**, the same with the forget gradient projected orthogonal to the mean gradient of the concurrent retain batch at every step, an unconditional rank-one instance of the projection family (Wang et al. (2025) apply the same correction only when the two gradients conflict; Shamsian et al. (2025) orthogonalize per sample); and **GA-prec**, plain SGD on the forget loss with the gradient divided coordinate-wise by $\sqrt{\mathcal{F}_{ii}} + \epsilon$, where $\mathcal{F}$ is the diagonal empirical Fisher of the retain evaluation subsample — a fixed 300 of the 3600 retain pairs, used for every evaluation in this batch — computed once at θ_0 and $\epsilon = 0.1$ times the root-mean-square of $\sqrt{\mathcal{F}_{ii}}$. GA-prec is Adam's update with the denominator taken from the retain side instead of the forget side, which is what the local model of §5 says should buy selectivity; it has to use SGD because Adam's own normalization would undo the preconditioning. Its learning rate was found by a ladder on seed 0 under the acceptance rule. Because the preconditioned gradient is clipped, GA-prec takes unit-length steps of size lr, so the registered grid (10^{-6} to 5×10^{-5}) was inert at its first rung and was raised to (10^{-4}, 3×10^{-4}, 10^{-3}) before any further rung was run, a deviation logged as an amendment: 10^{-3} is the first to enter the band (first in-band evaluation at step 504, accepted checkpoint at 580). The band arrives after 580–628 steps, against 18–34 for GA. All predictions and negative-result conditions were registered before the runs (amendment A-48).

Table 22: The fp32 batch at matched forget level. "cos pure" is against the purified oracle axis. Δretain and $\Delta F'$ are changes in answer-token NLL on the full retain sample (300) and F' (200), unrounded, in units of 10^{-4}; "frac" is the fraction of retain questions whose NLL rose, paired against M_0. The bf16 GA row is the registered run of Table 4; the oracle row is the point on the retrain path at the same forget level ("med." is the median over questions).

	step	$\|\Delta\theta\|$	cos pure	Δret. (10^{-4})	frac	$\Delta F'$ (10^{-4})
GA, bf16	506–566	0.69–0.75	0.0026–0.0028	1–2	—	—
GA, fp32	18–34	0.20–0.31	0.0083–0.0092	1.2–3.4	0.90–0.95	1.1–2.7
GA-proj, fp32	18–34	0.20–0.31	0.0084–0.0092	1.1–3.4	0.88–0.93	1.1–2.7
GA-prec, fp32	580–628	0.14–0.15	0.0051–0.0056	0.15–0.17	0.68–0.69	0.07–0.09
oracle, α=0.32	—	10.4	1	0.6 (med. −0.5)	0.42	−0.2

Precision. With fp32 weights every coordinate moves and the update is shorter, not longer: $\|\Delta\theta\|$ falls from 0.7 to 0.2–0.3 and the band arrives fifteen to thirty times sooner. The alignment rises from 0.003 to 0.0083–0.0092, which is the Llama value; the extra orthogonality of the second family in Appendix A was a property of bf16 rounding, not of the family. The steepness gap widens (the methods are 34–53× steeper than the oracle at matched level, against 15× in bf16), and retain NLL rises on 90–95% of retain questions. None of the four registered negative-result conditions fired: orthogonality and the steepness asymmetry are not artifacts of precision.

Projection. GA-proj is GA. The two updates of the same seed sit at cos = 0.998–0.999, every number in the table agrees to the third decimal, and the projection removed a median 0.006–0.02% (maximum 0.3%) of the forget gradient's squared norm: the forget gradient is already nearly orthogonal to the retain batch gradient, so there is nothing for a rank-one projection to remove. This is the direct form of the point made in §1 — the obstacle is not that the two gradients are entangled. Higher-rank or subspace projections (Wang et al., 2025; Shamsian et al., 2025; Zhou et al., 2025) can remove more; a separate diagnostic on these checkpoints (fp32, the last three transformer blocks) puts the forget gradient's energy inside the top-32 retain-gradient subspace at 1.4–2.1% over three seeds. That figure bounds how much of the gradient such a projection can remove, not how much it can matter: if the overlapping energy sat in the stiffest retain directions its removal could count for more than its share of the norm, and we have not measured that.

Preconditioning. GA-prec does what the local model predicts and nothing the paper did not already say. Its retain cost is an order of magnitude below GA's — paired per question, GA is worse on 83–94% of the retain set ($p < 10^{-30}$) — and F' is essentially untouched, so at matched forgetting it is the most selective update we ran, of the same order as the retrain path itself at that level (lower in the mean, higher in the median: the oracle path lowers most retain questions slightly and raises a few). And it is less aligned than GA: cos = 0.0051–0.0056 against the purified axis, about half of GA's, while being far more reproducible across seeds (cos = 0.91–0.93 between seeds, against 0.38–0.63 for GA). Selectivity was bought by moving along a different, well-defined direction, not by moving toward the retraining direction. Two cautions. The forgetting is band-level: forget ROUGE-L at the accepted checkpoints is 0.95–0.97, so this is not evidence that preconditioning removes the data, only that the local model's account of which direction is selective at a step is right. And the runs are on one family at one forget level; whether the preconditioned direction keeps its selectivity under the distance law of §7, which the oracle path escapes, is not tested here.

Does the distance law's exponent survive dense updates? §7 fits retain damage as α^p on bf16 Llama runs whose updates live on 7.3% of coordinates, so the exponent could be an artifact of that sparsity. We re-measured it with per-question fp32 evaluation on the registered grid, on the gradient-ascent runs of both families — so the comparison is method-matched, and narrower than the four-method fit in §7 (Table 23). It is not sparsity. Within the second family the bf16 and fp32 ranges overlap; what separates the numbers is the model family, Llama at 3.0–3.8 and Qwen at 1.5–2.2 depending on which α enter the fit. The claim that damage grows faster than quadratically is therefore a statement about the family in which we measured it, not a constant of the operation — which we have corrected in §7. What does hold everywhere is the weaker and load-bearing part: damage is monotone in distance, and no direction-based index predicts it.

Table 23: Retain-damage exponent p in $\Delta\mathrm{NLL}_R \propto \alpha^p$, three seeds each, fitted on the same grid with per-question fp32 evaluation. Points below the 10^{-5} evaluation floor are dropped, which for the bf16 runs leaves exactly the four $\alpha \geq 0.5$; "common" restricts every fit to those four, so the two precisions are compared on the same points.

family	precision	p (all α above floor)	p (common α)	$\|\Delta\theta\|$
Llama-3.2-1B	bf16	3.80, 2.99, 3.45	—	0.58–0.63
Qwen2.5-0.5B	bf16	2.00, 2.13, 2.08	same points	0.69–0.75
Qwen2.5-0.5B	fp32	1.64, 1.52, 1.45	2.20, 1.74, 1.63	0.20–0.31

A prediction of the local model, visible once the evaluation is precise enough. §5 notes that θ_0 minimizes the joint objective rather than L_R, so the first-order term in retain cost is negative: moving along ∇L_F should *lower* retain loss before curvature takes over. The rescaled Llama updates show exactly that. At $\alpha = 0.125$ the change in retain NLL is -1.5, -1.8 and -2.1×10^{-4} in the three seeds, and two of the three are still negative at $\alpha = 0.25$; the crossing to positive happens between $\alpha = 0.125$ and 0.5. The bf16 evaluation used in §7 put these points below its resolution rather than below zero, which is why the effect was not visible there.

N Depth: alignment by layer, and why layer mass does not explain it

Table 24 gives the by-layer version of the alignment reported in §4. Alignment with Δ_{oracle} rises with depth for every method (monotonically for GA and NPO; GradDiff and the task vector dip once between layers 0 and 1), and in layer 0 GA and NPO sit at the random floor: whatever the oracle direction does early in the network, they do not do it at all. Task-vector negation carries the same profile an order of magnitude higher, which is the by-layer form of Table 5.

Table 24: $\cos(\Delta\theta^{(l)}, \Delta^{(l)}_{\text{oracle}})$ computed within each layer's parameter block, mean over 3 seeds on the F arm. The random floor is $1/\sqrt{d}$: 2.8×10^{-5} for the whole model ($d = 1.24 \times 10^9$) and $\approx 1.3 \times 10^{-4}$ for one transformer block ($d \approx 6.1 \times 10^7$). Layer 0 is at that floor for GA and NPO.

Layer	GA	GradDiff	NPO	Task vector
0	−0.0001	+0.0018	−0.0001	+0.0429
1	+0.0008	+0.0015	+0.0009	+0.0186
2	+0.0051	+0.0038	+0.0051	+0.0461
3	+0.0061	+0.0051	+0.0062	+0.0489
4	+0.0071	+0.0062	+0.0073	+0.0495
5	+0.0081	+0.0072	+0.0083	+0.0515
6	+0.0084	+0.0075	+0.0087	+0.0531
7	+0.0091	+0.0087	+0.0094	+0.0588
8	+0.0105	+0.0090	+0.0109	+0.0676
9	+0.0123	+0.0107	+0.0127	+0.0761
10	+0.0143	+0.0121	+0.0148	+0.0840
11	+0.0150	+0.0123	+0.0155	+0.0894
12	+0.0152	+0.0127	+0.0157	+0.0913
13	+0.0166	+0.0139	+0.0171	+0.0975
14	+0.0171	+0.0152	+0.0178	+0.1059
15	+0.0171	+0.0155	+0.0178	+0.1140

The gradient is not a mass artifact. A monotone trend with depth invites the reading that the oracle direction "lives" in the deep layers. It does not follow, and the mass distribution says why not. Table 25 gives each direction's share of squared norm by layer. The oracle, the F' control and the seed-only null, which forgets nothing, distribute their mass identically to the first decimal, and removing the dead component from the oracle ($\Delta^{\perp}_{\text{oracle}}$) does not move the profile either. A distribution shared by a direction that forgets nothing is a signature of fine-tuning drift, not of forgetting. What differs between the oracle and an unlearning update is the direction *inside* each layer — which is what Table 24 measures — not how the update is spread across layers. The one place the profiles do part company is the embedding and final-norm block, where the unlearning updates put 1% of their mass against the oracle's 17%.

Table 25: Share of $\|\Delta\|^2$ by layer (%). Block -1 is the embedding and final-norm parameters. Norms are matched across the first three columns (21.81, 21.88, 21.95), so these are shape comparisons, not size comparisons.

Layer	oracle	control	noise	$\Delta^{\perp}_{\text{oracle}}$	GA	GradDiff
-1	17.00	16.91	16.69	16.77	0.88	1.23
0	5.54	5.43	5.29	5.54	6.85	6.36
1	17.00	17.28	18.69	16.89	17.40	10.02
2	4.53	4.54	4.49	4.55	5.49	6.07
3	4.41	4.41	4.36	4.46	5.47	6.02
4	4.43	4.44	4.41	4.45	5.67	6.13
5	4.58	4.60	4.55	4.59	5.73	6.21
6	4.63	4.64	4.61	4.67	5.87	6.30
7	4.66	4.66	4.60	4.68	5.98	6.26
8	4.77	4.78	4.70	4.80	5.78	6.19
9	4.53	4.52	4.43	4.55	5.52	5.88
10	4.49	4.46	4.36	4.51	5.37	5.83
11	4.31	4.29	4.17	4.34	5.10	5.61
12	4.21	4.18	4.08	4.22	4.90	5.56
13	3.99	3.97	3.87	4.01	4.75	5.45
14	3.71	3.69	3.60	3.73	4.64	5.44
15	3.23	3.21	3.12	3.26	4.60	5.44